

ur-iw-hnt at CheckThat! 2022: Cross-lingual Text Summarization for Fake News Detection

Hoai Nam Tran¹, Udo Kruschwitz¹

¹Information Science, University of Regensburg, Germany

Abstract

We describe our submission to the CLEF CheckThat! 2022 challenge. We contributed to Tasks 3A and 3B – multiclass fake news classification in English and German, respectively. Our approach incorporates extractive and abstractive summarization techniques by utilizing fine-tuned DistilBART and T5-3B. For cross-linguality, we use automatic machine translation to improve model inference. Our approved run for Task 3B was the official winner according to both F1 and Accuracy, with a fair margin to the second place. For Task 3A, we describe a wide range of models that we experimented with. While only one submission per team was permitted, we also describe the non-submitted setup that tops the leaderboard performance in this task.

Keywords

Fake news detection, BART, T5, extractive summarization, abstractive summarization, translation

1. Introduction

The distribution of fake news is not a new problem, but due to its scale, it has become an urgent social and political issue [1]. Here we understand *fake news* to be intentionally and verifiably false information with the purpose of deceiving its reader [2].

Task 3 of the CLEF 2022 CheckThat! Shared Task [3] focuses on Multiclass Fake News Classification with English (3A) and German (3B) test sets (reusing the previous year’s dataset for training). That is the task with our contribution.

The critical motivations for our work are as follows. We have seen transformer-based models becoming the basis for most state-of-the-art NLP applications, including a wide range of classification tasks, e.g., [4]. However, we acknowledge that there are restrictions on the input size in transformer models, which is why we are inspired by the findings that automatic summarization as a step towards cutting down long documents has been demonstrated to help identify fake news [5]. Finally, there are indications that automatic machine translation can help improve text classification, e.g., in our own work [6].

In this paper, we use transformer models for summarization and multiclass classification. Additionally, we use automatic machine translation for the German subtask to improve model inference. We conducted several experiments, but only one submission was allowed, and in the official leaderboard, we ended up being ranked 1st in Task 3B and 9th in Task 3A. Here we also


CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy

✉ Hoai-Nam.Tran@student.ur.de (H. N. Tran); Udo.Kruschwitz@ur.de (U. Kruschwitz)

ORCID 0000-0002-5503-0341 (U. Kruschwitz)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

discuss our non-submitted approaches for which post-competition results demonstrate that they would be ranked 1st in both 3A and 3B with a substantial margin over the leaderboard’s best performers. This paper describes our experiments in more detail.

To encourage reproducibility of experimental work, we make all models available via Hugging Face¹ and additional data via GitHub².

2. Related Work

We briefly sketch related work here and focus on what directly inspired us for this paper.

Fake News Detection is an unresolved task for which Transformer models with self-attention [7] like BERT [4], BART [8], and T5 [9] are actively utilized. Since fake news can appear in every language, multilingual models like XLM-RoBERTa [10] apply their cross-lingual ability in several tasks and benchmarks. One such dataset is FakeCovid, consisting of fact-checked articles from 92 fact-checking websites [11]. Shahi et al. [12] conducted an exploratory study of COVID-19 misinformation on the Twitter platform to define four different classes for the current dataset used in this Shared Task [3] and in the previous one [13]. To collect high-quality data, Shahi [14] proposes a semi-automatic framework where both machines and humans are involved in the process to mitigate the workload.

In last year’s Shared Task, Hartl and Kruschwitz used the same DistilBART model we adopt here for summarization (though we use it for extractive rather than abstractive summarization) [15]. In later work, they refined this approach to achieve state-of-the-art performance for the task of fake news detection using a common reference benchmark collection [5].

3. Dataset

The dataset has been annotated using four labels: "true", "false", "partially false", and "other". As indicated in Table 1, the distribution of the released dataset is rather imbalanced. The training set is the same as last year’s 3A task [13]. The difference here is the later released test sets consisting of 612 English data points for task 3A and 586 German data points for task 3B. Both test sets have substantially more "true" labels than "partially false", while the training set and development set have more "partially false" than "true" labels.

As shown in Table 2, the dataset contains some very long texts in both title and text. Therefore, one challenge of this Shared Task is to consider this length, especially when it goes beyond the standard token limits of typical transformer models.

¹https://huggingface.co/hntran/CLEF_2022_CheckThatLab_Task3

²https://github.com/HN-Tran/CLEF_2022_CheckThatLab_Task3

Table 1
Some Dataset Statistics

Labels	Training Set	Development Set	Test Set 3A	Test Set 3B
True	142	69	210	243
False	465	113	315	191
Partially False	217	141	56	97
Other	76	41	31	55
All	900	364	612	586

Table 2
Token Length Distribution

Variables	Statistics	Training Set	Development Set	Test Set 3A	Test Set 3B
Title	Median	70	66	73	67
	Mean	286	171	78	71
	Minimum	3	3	11	3
	Maximum	9960	8092	200	234
Text	Median	3035	3115	3655	4009
	Mean	4167	4498	6052	5617
	Minimum	18	25	289	507
	Maximum	32767	44359	100000	45309

4. Methodology

4.1. Summarization

For the summarization task, we use two particular models for two different approaches: DistilBART-CNN-12-6³ for extractive summarization and T5-3B⁴ for abstractive summarization.

Extraction-based summarization selects the best representations of words and sentences of the given document or text input. In contrast, abstraction-based summarization generates shorter text which can be new sentences capturing the prominent notion of the source text input. In the used library [16] (see Chapter 4.4), we extract the output embeddings from the chosen model inference and cluster these with k-means [17]. The Elbow method is used to determine the optimal k-value [18]. While it is also possible to restrict the amount of output text by a fixed number of sentences or a fixed ratio, we want to use the optimal cluster of sentences instead to avoid losing any possibly relevant sentences. Our chosen model is DistilBART-CNN-12-6 which is based on BART [8] with distillation [19], fine-tuned with the CNN and DailyMail dataset [20]. In contrast to extraction, we use the three billion parameter version of T5 [9] to generate shorter sentences with the same prefix used in the pre-training process for the CNN/DailyMail dataset [20]. While the default input token length limit is 512, the model uses relative positional embeddings, which allows it to utilize much longer text at the cost of higher

³<https://huggingface.co/sshleifer/distilbart-cnn-12-6>

⁴<https://huggingface.co/t5-3b>

computing resources like memory consumption [21, 9]. We also apply some omissible soft pre-processing steps to ensure retention of usually lossy token information and cleaner input for further processing. To solve the problem of token length, as mentioned in Chapter 3, we combine the title with the text for summarization in the following order: "title" + "." + "text".

We refer the interested reader to the project repository for further details.

4.2. Multi-Class Classification

The dataset has four different classes (see Table 1), which have imbalanced distributions. To classify them, we use BERT_{Base} Uncased [4] as our baseline model and include three large models: BERT_{Large} Uncased [4], XLM-RoBERTa_{Large} [10], and T5-3B [9]. The default fine-tuning process consists of tokenization, splitting the dataset into train, development, and dev-test sets, and then the actual training. After the automatic machine translation step for the cross-lingual task, the inference is the last step for classification of the test set for submission and for the later released ground-truth labels to see how well our fine-tuned models perform. The main metric is macro-F1 since the dataset is imbalanced; specifically, the averaged F1 score as in Equation 1 is calculated [22].

$$\mathcal{F}_1 = \frac{1}{n} \sum_x \text{F1}_x = \frac{1}{n} \sum_x \frac{2P_x R_x}{P_x + R_x} \quad (1)$$

4.3. Machine Translation

For the cross-lingual task (3B), we use the free Google Translate service to translate the whole German test set into English for inference. In our previous work, this has been shown to be effective [6]. Given the scale of translation data, Google utilizes this as an obvious choice. Since Google Translate has an internal character limit, we only take the first 5000 tokens for translation. After the automatic machine translation, we repeat the summarization step on these newly created data and start the inference process.

4.4. Experimental Setup

All of our experiments are conducted on a single RTX A6000 with 48 GB VRAM. We use the SimpleTransformers library⁵ for the T5 model and all other transformer models with the Hugging Face Transformers library⁶. For the summarization task, we use the Bert Extractive Summarizer library⁷ [16], and for machine translation, we use the deep-translator library⁸ in combination with the free public Google Translate service⁹.

⁵<https://simpletransformers.ai/>

⁶<https://huggingface.co/transformers>

⁷<https://github.com/dmmiller612/bert-extractive-summarizer>

⁸<https://github.com/nidhaloff/deep-translator>

⁹<https://translate.google.com/>

5. Experiments

First, we conduct a stratified split of the development set by the standard 80:20 ratio to have a dev-test set to choose our submission. Then in our experiments, we make five runs of each model with the default hyperparameters, including these changes for all models (except for T5-3B):

- Maximum Steps: 705
- Learning Rate: 1e-5
- Max Sequence Length: 256
- Batch Size: 32
- Warmup Ratio: 0.1
- Weight Decay: 0.01

We take the best model with the highest macro-F1 score after saving models every 50 steps. For the T5 model, we make one single run with the following changes from the default:

- Maximum Epochs: 200
- Max Sequence Length: 256
- Batch Size: 4
- Early Stopping Metric: Macro-F1
- Early Stopping Delta: 0.01
- Early Stopping Patience: 5

Table 3 shows that the T5-3B classifier with DistilBART-CNN-12-6 as the extractive summarizer is the best overall model for both tasks, with 39.54% (best in 3A) and 29.58% (second-highest in 3B), respectively. Our submission (marked with *) is the extraction-based BERT_{Large} model’s first run taking 1st place in the 3B leaderboard, and is the third-highest performer of our experiments for Test 3B. The best performing model in 3B is XLM-RoBERTa_{Large} with T5-3B as the abstractive summarizer with a macro-F1 score of 30.06%. For 3A, the best abstractive classification model is BERT_{Large} with 36.48%; for 3B, the best extractive classification model is T5-3B with 29.58%. All results are macro-F1 scores (see Equation 1).

6. Discussion

We observe that the variation of the different performances between the five runs of each model is high (up to 9.53% in extractive BERT_{Large} for Test 3B), an indication of overfitting. Some possible explanations for this might be the choice of imprecise hyperparameters or the substantial discrepancy between the different parts of the dataset. Furthermore, the abstractive T5-3B test result might also be caused by overfitting. While our submission is over the 50% mark and has the lowest difference of only 0.22%, the seemingly stable results between the development set and the dev-test set do not guarantee a good score in the test set. Interestingly, the extractive T5-3B has scored on the dev and dev-test set lower than 50% and is the best overall performer. Abstractive summarization generally gives the BERT models higher macro-F1 results than extractive summarization. Nevertheless, the results of both summarization techniques are similar, and thus both approaches are still viable.

Table 3

Experimental runs conducted for Tasks 3A and 3B (actual submission marked with *)

Summarization Model	Classification Model	Run Nr.	Dev	Dev-Test	Test 3A	Test 3B
DistilBART-CNN-12-6 (extractive)	BERT _{Base}	1	47.59	48.19	28.39	23.81
		2	50.02	44.27	28.48	27.22
		3	48.16	41.41	30.20	25.94
		4	49.97	41.47	29.88	26.21
		5	48.04	47.06	32.23	25.98
	BERT _{Large}	1*	52.40	52.18	28.33	28.99
		2	46.43	39.96	26.87	19.46
		3	48.77	52.78	30.70	28.69
		4	49.21	48.44	32.31	25.32
		5	53.25	51.85	30.19	20.46
	XLM-R _{Large}	1	50.53	41.04	30.42	27.40
		2	50.93	44.54	33.11	28.01
		3	49.08	48.56	30.82	26.09
		4	50.80	43.99	28.23	21.94
		5	50.95	40.29	32.47	23.34
T5-3B	1	48.05	46.52	39.54	29.58	
T5-3B (abstractive)	BERT _{Base}	1	54.05	45.76	35.41	27.14
		2	48.12	44.73	31.88	28.03
		3	50.02	40.58	33.73	25.84
		4	49.58	47.21	31.86	23.91
		5	48.89	40.29	31.13	24.18
	BERT _{Large}	1	56.33	51.15	28.89	21.34
		2	45.85	37.87	32.88	23.43
		3	55.08	46.80	35.24	28.33
		4	52.15	47.08	36.48	27.01
		5	51.32	46.91	30.56	21.77
	XLM-R _{Large}	1	51.54	44.81	31.66	28.99
		2	49.36	42.84	35.63	30.06
		3	49.73	44.91	35.67	27.82
		4	50.59	44.79	36.01	26.86
		5	51.78	40.25	35.29	28.09
T5-3B	1	52.08	43.82	29.72	23.72	

6.1. Limitations

Because of time constraints, we have not implemented ensembling strategies in our experiments. However, there is plenty of scope as ensembles have been demonstrated to offer substantial gains over individual classifiers, e.g., [5, 23].

For the same reason, only one summarization model for each technique and the number of transformer models for the classification task was possible. The chosen ratio of the stratified

split might cause too few data points for the dev-test set, so a different ratio like 50:50 would have been an alternative.

6.2. Future Work

In the future, it would be interesting to see how good larger models like XLM-R_{XL/XXL} [24] or other models like Big Bird [25] or PEGASUS [26] perform. For the cross-lingual task, using multilingual models without needing machine translation is another option to experiment with. Alternatively, text generation models like T5 [9] and BART [8] can also be used for machine translation.

7. Conclusion

We described our family of approaches to the task of multiclass fake news classification for English and German. At the core, they use fine-tuned transformer architectures and incorporate extractive and abstractive summarization (to be able to deal with long input documents). For the multilingual task, we also incorporate automatic machine translation. The results demonstrate that both summarization techniques and automatic machine translation are competitive. In particular, for the multilingual setting, we observe a large margin between our winning submission and the places further down on the leaderboard. Our analysis also uncovers that large language models perform best if overfitting can be avoided.

Acknowledgments

This work was supported by the project *COURAGE: A Social Media Companion Safeguarding and Educating Students* funded by the Volkswagen Foundation, grant number 95564.

References

- [1] P. Nakov, D. P. A. Corney, M. Hasanain, F. Alam, T. Elsayed, A. Barrón-Cedeño, P. Papotti, S. Shaar, G. D. S. Martino, Automated fact-checking for assisting human fact-checkers, in: Z. Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021, ijcai.org, 2021, pp. 4551–4558.
- [2] H. Allcott, M. Gentzkow, Social media and fake news in the 2016 election, *Journal of economic perspectives* 31 (2017) 211–36.
- [3] J. Köhler, G. K. Shahi, J. M. Struß, M. Wiegand, M. Siegel, T. Mandl, M. Schütz, Overview of the CLEF-2022 CheckThat! lab task 3 on fake news detection, in: Working Notes of CLEF 2022—Conference and Labs of the Evaluation Forum, CLEF ’2022, Bologna, Italy, 2022.
- [4] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational

- Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: <https://aclanthology.org/N19-1423>. doi:10.18653/v1/N19-1423.
- [5] P. Hartl, U. Kruschwitz, Applying automatic text summarization for fake news detection, in: Proceedings of the 13th Language Resources and Evaluation Conference (LREC 2022), 2022, pp. 2702–2713.
- [6] H. N. Tran, U. Kruschwitz, ur-iw-hnt at GermEval 2021: An ensembling strategy with multiple BERT models, in: Proceedings of the GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments, Association for Computational Linguistics, Duesseldorf, Germany, 2021, pp. 83–87. URL: <https://aclanthology.org/2021.germeval-1.12>.
- [7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17, Curran Associates Inc., USA, 2017, pp. 6000–6010. URL: <http://dl.acm.org/citation.cfm?id=3295222.3295349>.
- [8] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 7871–7880. URL: <https://aclanthology.org/2020.acl-main.703>. doi:10.18653/v1/2020.acl-main.703.
- [9] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, *Journal of Machine Learning Research* 21 (2020) 1–67. URL: <http://jmlr.org/papers/v21/20-074.html>.
- [10] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451. URL: <https://aclanthology.org/2020.acl-main.747>. doi:10.18653/v1/2020.acl-main.747.
- [11] G. K. Shahi, D. Nandini, FakeCovid – A Multilingual Cross-domain Fact Check News Dataset for COVID-19, in: Workshop Proceedings of the 14th International AAAI Conference on Web and Social Media, 2020. URL: http://workshop-proceedings.icwsm.org/pdf/2020_14.pdf.
- [12] G. K. Shahi, A. Dirkson, T. A. Majchrzak, An exploratory study of covid-19 misinformation on twitter, *Online Social Networks and Media* 22 (2021) 100104.
- [13] G. K. Shahi, J. M. Struß, T. Mandl, Overview of the CLEF-2021 CheckThat! lab task 3 on fake news detection, Working Notes of CLEF (2021).
- [14] G. K. Shahi, AMUSED: An Annotation Framework of Multi-modal Social Media Data, arXiv preprint arXiv:2010.00502 (2020).
- [15] P. Hartl, U. Kruschwitz, University of Regensburg at CheckThat! 2021: Exploring Text Summarization for Fake News Detection, in: CLEF (Working Notes), volume 2936 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021, pp. 508–519.
- [16] D. Miller, Leveraging bert for extractive text summarization on lectures, arXiv preprint arXiv:1906.04165 (2019).
- [17] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley symposium on mathematical statistics and

- probability, volume 1, Oakland, CA, USA, 1967, pp. 281–297.
- [18] T. M. Kodinariya, P. R. Makwana, Review on determining number of cluster in k-means clustering, *International Journal* 1 (2013) 90–95.
 - [19] S. Shleifer, A. M. Rush, Pre-trained summarization distillation, *ArXiv abs/2010.13002* (2020).
 - [20] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, P. Blunsom, Teaching machines to read and comprehend, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 28, Curran Associates, Inc., 2015. URL: <https://proceedings.neurips.cc/paper/2015/file/afdec7005cc9f14302cd0474fd0f3c96-Paper.pdf>.
 - [21] P. Shaw, J. Uszkoreit, A. Vaswani, Self-attention with relative position representations, in: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 464–468. URL: <https://aclanthology.org/N18-2074>. doi:10.18653/v1/N18-2074.
 - [22] J. Opitz, S. Burst, Macro f1 and macro f1, *arXiv preprint arXiv:1911.03347* (2019).
 - [23] S. Zimmerman, U. Kruschwitz, C. Fox, Improving hate speech detection with deep learning ensembles, in: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018, pp. 2546–2553.
 - [24] N. Goyal, J. Du, M. Ott, G. Anantharaman, A. Conneau, Larger-scale transformers for multilingual masked language modeling, in: *Proceedings of the 6th Workshop on Representation Learning for NLP (ReL4NLP-2021)*, Association for Computational Linguistics, Online, 2021, pp. 29–33. URL: <https://aclanthology.org/2021.repl4nlp-1.4>. doi:10.18653/v1/2021.repl4nlp-1.4.
 - [25] M. Zaheer, G. Guruganesh, K. A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang, A. Ahmed, Big Bird: Transformers for Longer Sequences, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), *Advances in Neural Information Processing Systems*, volume 33, Curran Associates, Inc., 2020, pp. 17283–17297. URL: <https://proceedings.neurips.cc/paper/2020/file/c8512d142a2d849725f31a9a7a361ab9-Paper.pdf>.
 - [26] J. Zhang, Y. Zhao, M. Saleh, P. Liu, PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization, in: H. D. III, A. Singh (Eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, PMLR, 2020, pp. 11328–11339. URL: <https://proceedings.mlr.press/v119/zhang20ae.html>.